

BAB II

LANDASAN TEORI

2.1 Game

Game yang berarti kegiatan yang memainkan permainan terdapat peraturan, *play* dan budaya dimana pemain bisa merasakan permasalahan buatan, disini pemain akan berhubungan langsung dengan sistem dan pertikaian yang ada pada permainan hasil rekayasa sistem.

2.1.1 Sejarah Game

Pertama kali muncul dan dibawa oleh seorang yang bernama Steven Russel dengan proyek bernama Computer Games tahun 1962 dengan judul Star Wars.

Di tahun 1970 an muncullah *game* yang sangat dikenal oleh kalangan *gamers* disaat itu, dengan judul *game* Pong dengan sistem disket atau cartridge. Kemudian di tahun 1980 an muncullah *game* populer disaat itu yaitu berbasis teknologi IBM PC yang bernama *game* Pacman. Sampai sekarang pacman masih populer dikalangan *gamers*.

Pada tahun 1993 terciptalah *game* Mortal Combat sebagai pesaing pacman. *Game* ini mengusung *genre* aksi. Lima tahun kemudian muncul *game* Doom dengan desain yang lebih canggih karena teknologinya telah menggunakan basis 3d tetapi sederhana dan sistem suara yang masih terbatas.

Perbedaan *game* tahun 90an *game* dianggap sebagai komoditas untuk anak-anak sedangkan pada era tahun 2000-an *game* sudah melampaui berbagai bidang seperti hiburan bagi semua orang mulai dari kalangan bisnis, simulasi, edukasi, dan juga pembelajaran virtual [6].

2.1.2 Genre Game

Genre atau ragam permainan video digunakan untuk menggolongkan permainan video berdasarkan interaksi bidang permainannya, bukan hanya perbedaan visual maupun naratif. Permainan video diklasifikasikan independen berdasarkan pengaturan atau isi dari sktruktur permainan tersebut, tidak seperti film ataupun buku.

Berikut adalah genre *game* menurut Andrew Rollings dan Ernest Adams [7].

a. *Action Games*.

Gambar 2.1 merupakan contoh *game* dengan genre *action games*.



Gambar 2.1 *Game Super Mario Bros*

Merupakan permainan aksi (*Action games*) yang memerlukan pemain untuk menggunakan refleks, akurasi, dan waktu yang tepat untuk menyelesaikan sebuah tantangan.

b. *Strategy Games*.

Gambar 2.2 merupakan contoh *game* yang membutuhkan pemikiran.



Gambar 2.2 *Age of Empires*

Merupakan permainan strategi yang berfokus pada pemikiran yang cermat, dan perencanaan yang matang untuk mencapai kemenangan. Contoh : *Age of Empires*

c. *Role Playing Games*.

Gambar 2.3 merupakan contoh *game* dengan genre *Role-Playing Games*.



Gambar 2.3 *Game Diablo*

Suatu permainan yang melibatkan seorang player untuk berperan sebagai seorang “*Adventures*” atau petualang yang mengkhususkan diri dalam keahlian tertentu (seperti pertempuran jarak dekat, pertempuran jarak jauh. Dan keahlian mantra atau sihir).

Sementara itu pemain menjalani *storyline* / alur cerita yang telah ditentukan dalam *game*. Sub-genre dari *Role-Playing Games* juga telah banyak berkembang saat ini, misalkan :

- a. *Action RPG*, suatu jenis *game RPG* yang menggabungkan unsur *action* atau *action-adventure*. Contoh : *Diablo*
- b. *MMORPG (Massively Multiplayer Online Role Playing Games)*



Gambar 2.4 *Game Kingdom Under Fire 2*

Penjelasan gambar 2.4 merupakan Suatu permainan Online RPG yang tidak hanya melibatkan seorang player, tetapi melibatkan ratusan sampai ribuan pemain dalam satu waktu dalam menyelesaikan misi dan juga saling membantu dan memperkuat satu sama lainnya. Contoh : *Kingdom Under Fire 2*

c. *Tactical RPG.*



Gambar 2.5 *Game Final Fantasy Tactic*

Pada gambar 2.5 merupakan suatu permainan dimana pemain mengontrol beberapa pihak dan melakukan pertempuran melawan pihak musuh yang mempunyai jumlah yang sama, genre ini menggabungkan *gameplay* strategis seperti gerakan taktis di grid isometric, Contoh : *Final Fantasy Tactic*

d. *Fantasy.*

Gambar 2.6 merupakan contoh *game* yang memiliki nilai khayalan yang tinggi.



Gambar 2.6 *Game Star Ocean*

Suatu permainan yang biasanya mempunyai pengaruh dari abad pertengahan dengan tingkat fantasy (khayalan) yang tinggi. Contoh : [Star Ocean](#) series , *Final Fantasy series*.

e. *Sports Game.*



Gambar 2.7 *Pro Evolution Soccer 2017*

Pada gambar 2.7 merupakan suatu permainan yang mensimulasikan olahraga, dimana tim lawan dapat dikendalikan oleh pemain lain atau Kecerdasan buatan (AI). Sub-genre dari *Sports Game* seperti *Racing Games* (Balapan). Contoh : *Pro Evolution Soccer* , *Nascar Series*.

f. *Vehicle Simulation.*

Pada gambar 2.8 merupakan contoh *game* yang bergenre simulasi.



Gambar 2.8 *Flight Simulator 2015*

Merupakan suatu permainan yang mencoba memberikan pemain dengan interpretasi realistis atau operasi nyata dalam mengendarai bermacam-macam kendaraan mulai

dari motor, mobil, truck, perahu dan pesawat terbang Contoh : *Flight Simulator*, *Train Simulator*, *Bike Simulator*, *Bicycle Simulator*, *Truck Simulator*.

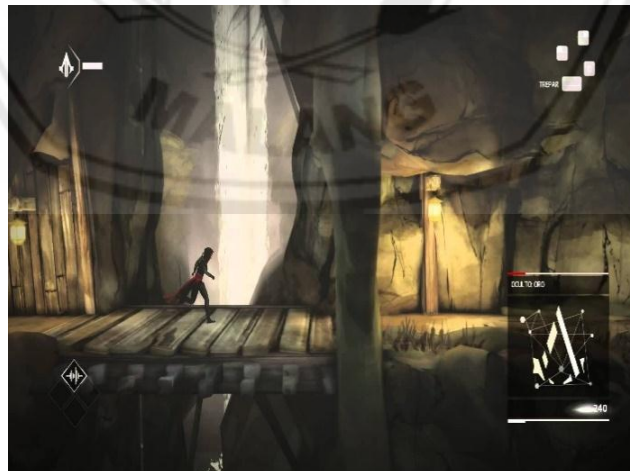
g. *Construction and Management Simulations*.



Gambar 2.9 Game SimCity

Merupakan suatu permainan yang memberikan interpretasi realistis yang hampir sama dengan *Vehicle Simulations*, tetapi dalam *Construction and Management Simulations* pemain mempunyai tugas untuk membangun, memperluas, atau mengelola sumber daya, proyek serta komunitas fiksi dengan sumber daya yang terbatas. Contoh : *SimCity*, *Cities SkyLines*, *Cities XXL*.

h. *Adventure Games*.



Gambar 2.10 Assassin's Creed Chronicles China

Pada gambar 2.10 merupakan suatu permainan petualangan yang bukan kompetisi dan simulasi dimana permainan ini tidak menawarkan proses untuk dikelola atau

mengalahkan seorang lawan menggunakan taktik/strategi , melainkan sebuah *game* yang dihadapkan kepada pemain untuk melakukan suatu petualangan yang mempunyai cerita interaktif tentang seorang karakter dalam sebuah *game*. Contoh : *Assassin's Creed*, *The Last of Us*, *Resident Evil*.

i. *Artificial Life, Puzzle Games*



Gambar 2.11 *Game Candy Crush Saga*

Pada gambar 2.11 merupakan suatu permainan yang merupakan cabang dari penelitian ilmu computer, Sedangkan *Puzzle Games* merupakan suatu permainan tentang bagaimana memecahkan sebuah teka-teki, dan biasanya teka-teki tersebut dijadikan sebagai bagian alur cerita dari suatu *game* yang harus dipecahkan oleh seorang pemain. Contoh : *Candy Crush Saga*, *Farm Heroes Saga*, *Cut The Rope*, *Pou*

j. *Online Games.*



Gambar 2.12 *Game Dota 2*

Pada gambar 2.12 merupakan suatu permainan yang dapat dimainkan secara parallel melalui jaringan Internet atau jaringan computer lainnya (LAN,WLAN). Saat ini

Online Games telah jauh berkembang daripada dahulu, seperti MMORPG (*Massively Multiplayer Online Role-Playing Games*), MOBA ([*Multiplayer online battle arena*](#)). Contoh : *World of Warcraft*, *Dota 2* , *League of Legends*

2.1.3 Game Edukasi

Dalam *Game* memiliki fungsi dan berdampak positif untuk anak, diantaranya, anak di ajarkan tentang teknologi komputer, belajar untuk diarahkan dan mengikuti aturan, berlatih untuk memecahkan masalah dan berpikir memakai logika, melatih saraf saraf motoric, saling komunikasi antara anak dan orang tua saat bermain dan berkumpul bersama, juga memberikan efek hiburan. Edukasi adalah proses pembelajaran dalam menemukan jati diri seseorang, dengan cara mengamati dan belajar yang kemudian lahirlah suatu pikiran tindakan. Edukasi tidak memiliki perbedaan dari belajar yang dikembangkan dalam ilmu psikologi [8].

2.1.4 RPG (Role Playing Game)

Permainan jenis ini pada umumnya memainkan pemeran atau tokoh. Biasanya pemain akan menjalankan sebuah cerita yang terdapat pada *game*. *Genre* RPG ini paling sering dimainkan oleh penduduk Indonesia. Sehingga apabila orang memainkan permainan yang berunsur ini maka pemain akan merasakan seperti menjadi pemeran yang terdapat pada permainan ini. Adapun jenis-jenis *game* RPG sebagai berikut: [9].

a. Fantasy

Kategori ini berseting rata-rata di dunia pertengahan Eropa dan menggunakan sihir, serta agama yang berbeda dengan dunia nyata. Contoh: *Final Fantasy* dan *Lord Of The Ring*.

b. Science Fiction

Ini terbalik dari fantasi RPG, setingan justru ada di zaman yang jauh lebih maju dari zaman sekarang. Sihir di ganti dengan istilah semacam *Telekinesis* atau *Psy*. Contoh: *Star Wars* dan *Enter the matrix*.

c. Historical

RPG ini lebih menekankan aspek sejarah, walaupun sering tertukar dengan Fantasy RPG. Yang membedakan, RPG ini mengambil setting di dunia nyata, namun di masa lalu. Contoh: *Age of Empire*.

2.1.5 Game Development Life Cycle (GDLC)

Game development life cycle adalah sebuah metode pembangunan dan pengembangan game, dimana di dalamnya ada tahapan initiation, pre-production, production, testing, beta, dan release, berikut penjelasan singkat tentang tahapan-tahapan game development life cycle yang bisa disingkat GDLC [10].

a. Initiation

Pada tahap ini akan dilakukan pembuatan konsep game yang ingin dibangun, akan seperti apa game itu dibangun. Pada Initiation akan menghasilkan konsep game dan deskripsi sederhana game.

b. Pre-Production

Tahapan sebelum production yaitu tahapan untuk melakukan perancangan dan membuat desain game.

c. Production

Pada tahap ini adalah tahap produksi atau realisasi pembuatan game yang sudah di desain pada tahap Pre-production. Pembuatan asset game seperti core asset, GUI, dan music yang dibutuhkan di dalam game serta integrasi asset dan source code agar menjadi game yang bisa dimainkan.

d. Testing (*Alpha*)

Pada tahapan ini akan dilakukan pengujian pada game yang sudah dibuat. Pengujian dilakukan oleh internal tim untuk menguji, game diuji pada lingkungan sebenarnya tetapi masih di lingkungan developer. Lingkungan dibuat sedemikian rupa agar menyerupai lingkungan sebenarnya. Hal ini dilakukan agar seluruh modul yang

ada dalam game tersebut berjalan sesuai aktifitas dan lingkungan sebenarnya. Pada testing alpha ini akan menghasilkan laporan bug, pada hasil ini nantinya akan memutuskan apakah game sudah siap untuk memasuki beta testing atau mengulangi ke tahap pre-production.

e. Beta

Beta adalah pengujian lanjutan setelah testing yang diuji oleh internal tim dan sekarang akan di uji oleh external atau diluar tim. Tujuan dari pengujian beta adalah untuk menempatkan game di tangan pengguna yang sebenarnya untuk menemukan setiap kekurangan atau masalah dari perspektif pengguna akhir. Apabila game sudah berjalan dengan baik maka dapat lanjut ke tahap berikutnya, tetapi apabila belum maka bisa kembali ke tahap pre-production.

f. Release

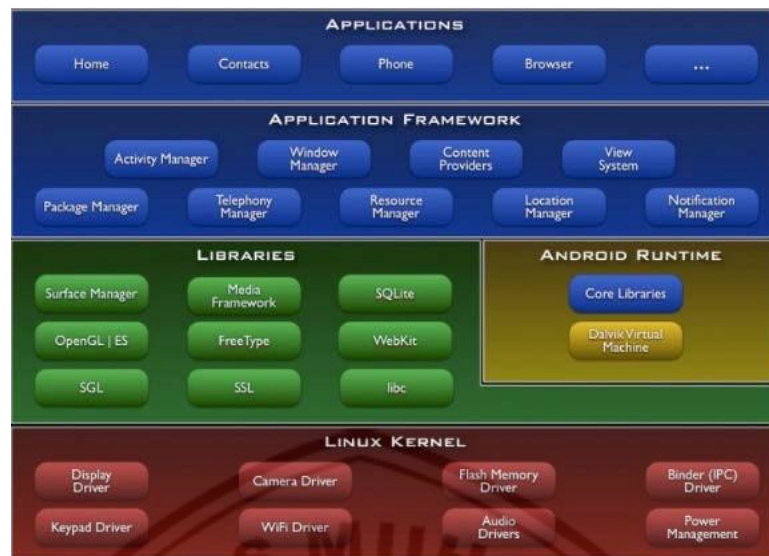
Pada tahap ini pembuatan game sudah mencapai tahap akhir dan berarti game sudah siap di publish ke publik. Game sudah resmi diluncurkan serta mengakhiri pembuatan game dan membuat dokumentasi pengembangan.

2.2 Lingkungan Pengembangan

2.2.1 Android

Android adalah sistem operasi terbuka berbasis Linux yang dirancang untuk perangkat selular layar sentuh seperti smartphone (telepon pintar) dan komputer tablet. Android awalnya dikembangkan bukan untuk perangkat seluler melainkan untuk system operasi canggih kamera digital, tetapi karena pasar tersebut tidak begitu besar maka pengembangan dialihkan bagi pasar ponsel pintar untuk menyaingi Symbian dan Windows mobile (saat itu iPhone belum dirilis).

Pada gambar 2.13 merupakan contoh komponen utama sistem android.



Gambar 2. 13 Komponen Utama Sistem Android

Sistem Operasi Android memiliki komponen utama sebagai berikut :

a. Aplikasi

Android pada umumnya menggunakan bahasa pemrograman Java, Contohnya *email client, SMS, kalender, dll.*

b. Kerangka kerja aplikasi

Biasanya aplikasi yang digunakan bahasa pemrograman java, sms, daftar kontak

Pada dasarnya, kerangka kerja aplikasi memiliki beberapa komponen sebagai berikut:

a. *Activity Manager*

Untuk mengatur navigasi yang dijalankan pada suatu proses yang berbeda.

b. *Package Manager*

Medeteksi aplikasi yang telah berada pada perangkat.

c. *Windows Manager*

Merupakan abstraksi dari bahasa pemrograman Java pada bagian atas dari level *services* (pada level yang lebih rendah) yang disediakan oleh *Surface Manager*.

d. *Telephony Manager*

Berisi sekumpulan API yang diperlukan untuk memanggil aplikasi.

e. *Content Providers*

Digunakan untuk memungkinkan aplikasi mengakses data dari aplikasi lain (seperti *contacts*) atau untuk membagikan data mereka sendiri.

f. *Resource Manager*

Digunakan untuk mengakses sumber daya yang bersifat bukan *code* seperti *string* lokal, *bitmap*, deskripsi dari *layout file* dan bagian eksternal lain dari aplikasi.

g. *View System*

Digunakan untuk mengambil sekumpulan *button*, *list*, *grid*, dan *text box* yang digunakan di dalam antarmuka pengguna.

h. *Notification Manager*

Berfungsi mengatur pemberitahuan dan kegunaan.

i. *Libraries*

Kemampuan ini bisa dilihat oleh berbagai pengembang melalui aplikasi yang berjalan [11].

2.2.2 Unity 2D

Perangkat lunak yang sifatnya membantu dalam pembuatan, pengembangan suatu permainan yang biasanya disebut *game engine*. Dimana *game engine* ini apabila digunakan lebih ekonomis, karena cukup dengan satu *game engine* mampu membuat banyak permainan dengan jenis yang berbeda-beda. Yang sering dipakai contohnya ialah *Unity*.

Unity dikembangkan dari sistem operasi OS X menggunakan bahasa pemrograman C++. Tahun 2008 adalah awal mula kejayaan Unity, ketika Apple meluncurkan iPhone App Store, Unity merupakan contoh media *game engine* pertama yang berusaha mengembangkan iPhone.

Unity mempunyai beberapa kelebihan antara lain:

GUI (*Graphical User Interface*) class dari Unity sangat cepat untuk digunakan, karena satu baris kode sudah dapat digunakan untuk memunculkan gambar.

- a. Kerapatan *pixel* dari gambar yang dihasilkan Unity sempurna.
- b. Pengguna dapat membagi tugas dalam pengerjaan sesuai bagian masing-masing, dan nanti hasil dari masing-masing pekerjaan dapat digabung menjadi satu kesatuan.

- c. Akses mudah ke *physics engine* milik Unity.
- d. Unity mampu menyebarkan permainan yang telah dibuat oleh pengguna ke berbagai perangkat seperti Windows, Mac, dan Android.
- e. Permainan secara otomatis dapat dijalankan pada berbagai resolusi layar yang tersedia.

Selain itu, Unity juga mempunyai beberapa kekurangan antara lain:

- a. Penggunaan *physics engine* tidak mudah, sehingga pengguna harus mempelajari dulu.
- b. Butuh baris kode tambahan agar permainan dapat dijalankan di berbagai resolusi layar.
- c. *Sprite manager* pada Unity lamban saat digunakan karena banyaknya baris kode dan pengaturan.

C# adalah bahasa pemrograman berorientasi objek dan berorientasi komponen, disini corel draw Merupakan sebuah aplikasi objek vector yang sering kita jumpai biasanya digunakan untuk membuat teks dan logo, salah satu contoh yang menggunakan format vector yaitu Corel Draw, freehand, dan lain-lain

2.3 Roti

Roti ialah makanan yang telah terdapat sejak zaman dahulu kala, bahkan sebelum dimulainya budaya bercocok tanam. Dengan mengumpulkan berbagai macam sumber daya dari alam sekitar, manusia mulai mengolah bahan-bahan tersebut agar dapat digunakan sebagai bahan pemenuhan utama, yaitu makanan. Contohnya: manusia mengumpulkan berbagai macam tanaman seperti gandum, lalu menyangrai gandum tersebut, yang kemungkinan besar menggunakan tepibatu yang panas karena terkena paparan sinar matahari agar dapat memisahkan biji dengan kulitnya. Hasil gandum ini lah yang digunakan oleh manusia zaman dahulu untuk membuat roti.

2.3.1 Roti Tawar

Roti tawar adalah roti yang dibuat melalui tahapan-tahapan yang bisa disebut adonan, menggunakan gula, susu dan ragi. Pada umumnya roti ini menggunakan

adonan yang tidak memakai telur mencampurkan atau tidak memakai gula, gula hanya dipakai pada proses pengembangan. Roti tawar memiliki tekstur yang halus seperti kapas, ringan dan rasanya tawar.(Husien, 2013:1).

2.3.2 Roti Manis

Roti yang terbuat dari tepung terigu dan dilakukan proses ragi roti, dipanggang. Biasanya boleh ditambah gula, garam, gula, susu, lemak dan selai cokelat, stroberi, dll.

2.3.3 Roti Basah

Merupakan Roti yang memiliki tekstur lebih lembut, tetapi roti basah biasanya tidak bertahan lama dikarenakan pada umumnya berisi santan atau ketan.

2.3.4 Bahan dan Alat Pembuatan Roti Secara Umum

a. Bahan :

Tepung Terigu, Mentega, Gula, Garam, Telur, Ragi, Susu Kental, Susu Cair, Vanili, Selai(Cokelat, Stroberi, Vanilla, kacang dll).

b. Alat :

Mixer, Timbangan Makanan, Takaran, Oven, Loyang, Mangkuk, Sendok Makan, dll.

2.3.5 Tahapan Secara Umum dalam Pembuatan Roti

a. Tahap 1 :

Menghitung (menimbang) bahan-bahan sesuai dengan yang dibutuhkan, Selanjutnya mempersiapkan semua peralatan dan perlengkapan yang dibutuhkan, Kemudian mengaduk semua bahan sehingga menjadi adonan yang kalis berguna untuk meratakan ragi hingga keseluruhan bagian adonan. Lalu mengembangkan gluten sehingga adonan melekat (tidak kering atau berderai).

b. Tahap 2 :

Proses pendiaman adonan dimana ragi akan bereaksi terhadap kandungan gula dan tepung, proses dari ragi tersebut akan menghasilkan CO₂ (Carbon Dioksida) dan Alkohol

c. Tahap 3 :

Meratakan adonan untuk mengeluarkan CO₂, mendistribusikan ragi kembali, sekaligus mengistirahatkan gluten, meratakan suhu dalam adonan.

d. Tahap 4 :

Membagi adonan menjadi potongan kecil dengan ukuran yang sama atau membagi adonan sesuai dengan jumlah yang dibutuhkan, bentuk adonan yang telah ditimbang sehingga berbentuk bulat, biarkan atau letakkan adonan diatas meja untuk mengistirahatkan gluten, biarkan adonan kembali mengembang karena proses fermentasi.

e. Tahap 5 :

Bentuk adonan sesuai dengan bentuk yang diinginkan, letakan adonan diatas Loyang atau didalam cetakan agar terbentuk selama proses pemanggangan, merupakan proses pengembangan terakhir sebelum dipanggang, pemanggangan sesuai dengan waktu dan suhu yang disesuaikan letakkan ditempat yang bersih dan memadai untuk mencegah roti menjadi rusak dan jangan lupa membungkus roti dengan kemasan plastic agar tidak cepat rusak [12].

2.4 Heuristic Evaluation

Evaluasi Heuristik adalah suatu cara pemeriksaan usability untuk perangkat lunak komputer yang membantu mengenali masalah-masalah usability pada rancangan antar muka (Jacob Nielsen dkk). Ada 10 prinsip aturan heuristik yang dikemukakan oleh Jacob Nelson, yaitu :

- a. **Tampilan Informasi dari status sistem;** prinsip ini berkaitan langsung dengan beberapa pertanyaan seperti : “Dimana saya sekarang?” dan “Dimana saya sekarang?” dan “Dimana jalan selanjutnya?”. Berdasarkan keterangan itu.
- b. **Komunikasi dengan pengguna;** sistem dapat berkomunikasi dengan user dengan bahasa yang dimengerti oleh user, dapat melalui istilah yang mudah dipahami oleh pengguna sehingga pengguna tidak mengalami kesulitan dalam menggunakan sebuah aplikasi.

- c. **Kontrol menu dari sistem;** Sistem menyediakan kontrol menu aplikasi yang mudah
- d. **Pencegahan error;** pencegahan pengguna ketika melakukan kesalahan, juga disediakan opsi konfirmasi sebelum pengguna melanjutkan aksi tersebut.
- e. **Konsistensi dan Standar;** agar suatu sistem menjadi standar, konsisten dalam hal penulisan kalimat, dan jenis huruf dan lain-lain.
- f. **Kemampuan untuk mengingat simbol;** aksi dan pilihan sistem harus terlihat oleh pengguna dan mudah untuk diakses. Sehingga pengguna tidak harus mengingat kembali informasi harus memanggil simbol.
- g. **Fleksibel dan Efisien Dalam Penggunaannya;** sistem harus dapat dibuat fleksibel dan seefisien mungkin baik untuk digunakan pengguna baru atau pemula.
- h. **Estetika dan minimalis desain;** sistem dialog dibuat secara estetis dan minimalis desain.
- i. **Membantu Pengguna Mengakui, Mediagnosa dan Memulihkan Sistem dari Kesalahan;** Sistem harus dapat memberikan pesan kesalahan/ error message dalam bahasa sederhana. Pesan kesalahan harus dijelaskan dalam bahasa yang jelas bukan dalam bentuk kode.
- j. **Bantuan dan Dokumentasi;** sistem harus disertai dengan informasi pengguna dan dokumentasi lain yang berhubungan dengan sistem. Informasi yang dibutuhkan user harus dapat dicari dengan mudah [13].

2.5 Playability Heuristic untuk mobile game.

Game berbeda dari perangkat lunak yang biasanya ada dalam sebuah karakteristik. Didalam permainan ini, tujuannya untuk kesenangan semata dalam bermain. Belajar untuk memainkan permainan, mengatasi masalah, atau menemukan hal baru yang merupakan bagian dari pengalaman. Disini pemain tidak mengetahui cara untuk mengembangkan dugaannya. Desain permainan ini dibuat dengan konten permainan dan menemukan pencapaian pemain. Memainkan permainan ini memang tidak sebenarnya. Tapi permainan ini menantang dan pemain perlu bekerja keras untuk mencapai tujuannya. Setelah itu, *usability heuristics* secara umum didalam evaluasi

permainan ini tidak lah cukup dan meninggalkan banyak aspek penting yang tidak dapat diperoleh dalam permainan. Kategori game heuristic ditegaskan sebagai berikut; game play adalah kesatuan masalah dan tantangan pemain untuk memenangkan game, game story ialah termasuk semua alur dalam cerita dan pembangunan karakter, game mechanics melibatkan sistem pemrograman yang menyediakan struktur oleh unit yang mempengaruhi lingkungan game atau biasa disebut latar didalam game, game usability membahas hubungan dan unsur antara pemain dan game meliputi mouse, keyboard, controller, game shell, heads-up display[14]. Kemudian riset tentang, *playability heuristic* untuk *game mobile* dilakukan oleh Samrad Soomro. Dalam riset ini daftar pendekatan untuk evaluasi *mobile game* tidak terlalu banyak dan *playability heuristic* ini dapat mendefinisikan masalah *playability* yang tidak dibahas pada *playability heuristic* sebelum-sebelumnya. Dengan daftar *heuristic* yang dibuat diharapkan dapat membantu para *game designer* dan *game developer* selama pengembangan game [15].